CLAIMS:

1. A method of computer data analysis using neural networks, the method including:

generating a data representation using a data set, the data set including a plurality of attributes, wherein generating the data representation includes:

modifying the data set using a training algorithm, wherein the training algorithm includes growing the data set; and

performing convergence testing, wherein convergence testing checks for convergence of the training algorithm, and wherein the modifying of the data set is repeated until convergence of the training algorithm occurs; and

displaying one or more subsets of the data set using the data representation.

2. A method according to claim 1, further including generating the data set using input data, and wherein generating the data set includes formatting the input data and initializing the formatted input data.

3. A method according to claim 2, wherein formatting the input data further includes creating a container class including a list of data vectors, $\overline{\mathbf{D}}$, where $\mathbf{d}_i$ is the ith vector in $\overline{\mathbf{D}}$, and $\mathbf{d}_{i,j}$ is the jth element of vector i.

4. A method according to claim 2, wherein formatting the input data further includes data scaling and binarisation of at least a portion of the data set.

5. A method according to claim 4, wherein data scaling includes replacing each element in each data vector in the data set by a scaled representation of itself, where:

$\forall i \in [1, card(\mathbf{d})]$, $\forall \mathbf{d}_i \in \overline{\mathbf{D}}$

$$d_{i,j} = \left(d_{i,j} - i_{\min}\right) \big/ \left(i_{\max} - i_{\min}\right).$$

6. A method according to claim 4 or 5, wherein binarisation includes converting attributes into one or more toggled attribute values.

7. A method according to any one of the preceding claims, wherein performing convergence testing includes testing condition $q(t) < Q_e$.

8. A method according to claim any one of claims 2 to 7, wherein initializing the formatted input data includes:

calculating an autocorrelation matrix, $\aleph$ over the input data set $\overline{\mathbf{D}}$,

where $\aleph = \frac{1}{card(\overline{\mathbf{D}})} \sum_{\forall \mathbf{d} \in \overline{\mathbf{D}}} \mathbf{d} \cdot \mathbf{d}^T$;

finding two longest eigenvectors of $\aleph$, $\mathbf{e}_1$ and $\mathbf{e}_2$, where $|\mathbf{e}_1| > |\mathbf{e}_2|$; and initializing vector values of each element of the data set F by spanning it with element values of the eigenvectors.

9. A method according to claim 8, wherein initializing the vector values includes:

$$F_{<F_R,0>} := \mathbf{0}$$

$$F_{<F_R,F_C>} := \mathbf{e}_1$$

$$F_{<1,F_C>} := \mathbf{e}_1 + \mathbf{e}_2$$

$$F_{<F_R,F_C>} := \mathbf{e}_2$$

$$\forall c \in [2, F_C - 1],\ F_{<1,c>} := \frac{F_C}{F_C - c} F_{<1,F_C>} + \frac{F_C - c}{F_C} F_{<1,1>}$$

$$\forall c \in [2, F_C - 1],\ F_{<R_C,c>} := \frac{c}{F_C} F_{<F_R,F_C>} + \frac{F_C - c}{F_C} F_{<F_R,1>}$$

$$\forall r \in [2, F_R - 1],\ F_{<r,1>} := \frac{c}{F_R} F_{<F_R,1>} + \frac{F_R - r}{F_R} F_{<1,1>}$$

$$\forall r \in [2, F_R - 1],\ F_{<r,F_C>} := \frac{r}{F_R} F_{<F_R,F_C>} + \frac{F_R - r}{F_R} F_{<r,F_R>}$$

$$\forall r \in [2, F_R - 1],\ \forall c \in [2, F_C - 1],\ F_{<r,c>} := \frac{c}{F_C} F_{<r,F_C>} + \frac{F_C - c}{F_C} F_{<r,1>}.$$

10. A method according to any one of the preceding claims, wherein the data set includes a plurality of data set nodes, and wherein growing the data set includes:

finding K_q for each of the data set nodes, where K_q is the node with the highest average quantization error, $\arg \max_q \{\overline{q}(t)_{K_q}\}$ for

each of the data set nodes, where $\bar{q}(t)_{K_q} = \frac{1}{t-1}\sum_{i=1}^{i=t-1} q(t)_{K_q}$ is the average quantization error for node q, where:

$$K_x = \arg\max_x\{\|K_q - K_{<r(q)-1,c(q)>}\|, \|K_q - K_{<r(q)+1,c(q)>}\|\}$$

$$K_y = \arg\max_y\{\|K_q - K_{<r(q),c(q)-1>}\|, \|K_q - K_{<r(q),c(q)+1>}\|\}$$

if $\|K_y - K_c\| < \|K_x - K_c\|$ then

$n_r = r(y)$ if $r(y) < r(c)$, else $n_r = r(c)$; and

$n_c = c(y)$;

else $n_r = r(y)$; $n_c = c(x)$ if $c(x) < c(c)$, else $n_c = c(c)$;

inserting a new row and column after row n_r and column n_c; and

interpolating new attribute values for the newly inserted node vectors using: $K_{<r,n_c>} = (K_{<r,n_c-1>} + K_{<r,c_n+1>})\frac{\alpha}{2}$ and $K_{<n_r,c>} = (K_{<n_r-1,c>} + K_{<n_r+1,c>})\frac{\alpha}{2}$, where $\alpha \in U(0,1)$.

11. A method according to any one of the preceding claims, wherein the training algorithm further includes:

$t = t + 1$;

$\forall \mathbf{d} \in \overline{\mathbf{D}}$;

if ($t < 50$ or $afterGrow$)

$$\wp_{\mathbf{d}} = \underset{<r,c>,\forall r\in[1,F_R],\forall c\in[1,F_C]}{\arg\min} \|\mathbf{d} - F_{<r,c>}\|_\rho$$

$afterGrow = false$

else

$\wp_{\mathbf{d}} = FindSCWS(\mathbf{d})$

call function: $FindNeighborhoodPatterns(\overline{\wp})$

call function: $BatchUpdateMatchVectors$

$$q(t) = \frac{1}{card(\overline{\mathbf{D}})}\sum_i (\|\mathbf{d} - F_{\wp\mathbf{d}}\|\rho);$$ and

if ($MayGrow(t)$ and $t < t_{\max}$), call function: $GrowKF$.

12. A method according to any one of the preceding claims, wherein displaying one or more subsets of the data set includes using a composite view

to view multiple attributes, wherein an additional attribute image is created, the additional attribute image displaying a union of a selected set of attributes.

13. A method according to claim 12, wherein using a composite view further includes:

constructing an attribute matrix; and

selecting a highest value for each attribute value from the selected set of attributes.

14. A method according to any one of the preceding claims, wherein displaying one or more subsets of the data set includes uses a range filter to select regions on the data representation and filter out nodes based on defined value ranges .

15. A method according to any one of the preceding claims, wherein displaying one or more subsets of the data set includes using a zooming function, wherein the zooming function includes:

making a selection of nodes to form a base reference of interest;

defining a set of data records from a second data set;

matching the second data set to the data representation;

flagging all records that are linked to the matched region; and

generating a second data representation using the flagged records.

16. A method according to any one of the preceding claims, wherein displaying one or more subsets of the data set includes using visual scaling, changing the minimum and maximum values used to calculate a colour progression used to visualize at least one of the plurality of attributes, and re-interpolating the active colour ranges over the new valid range of attribute values.

17. A method according to any one of the preceding claims, wherein displaying one or more subsets of the data set includes using a labeling engine to:

linking attribute columns in an input file to attributes in the data representation;

selecting attributes from the input file to be used for labelling;

determining with which row and column each row in the input file is associated; and

placing labels on the data representation.

18. A method according to any one of the preceding claims, wherein displaying one or more subsets of the data set includes using an advanced search function to:

read a set of data records from a data source;

match attribute columns from the set of data records to attributes in the data representation; and

display a list of all records that are associated with nodes that are part of the active selection on the data representation.

19. A method according to any one of the preceding claims, wherein displaying one or more subsets of the data set includes using equal distance averaging (EDA), wherein equal distance averaging includes:

finding the node vector, $\mathbf{n}$, in the data representation that most closely represents the input data vector, $\mathbf{d}$:

$\mathbf{n} = \arg\min_{K_{<r,c>}}\left\{\left\|K_{<r,c>} - \mathbf{d}\right\|_{\rho}\right\}, \forall r \in [1, K_R], \forall c \in [1, K_C]$; and

replacing missing entries in $\mathbf{d}$ with the corresponding entries from $\mathbf{n}$.

20. A method according to claim 19, wherein the equal distance averaging further includes:

building a list of the data representation nodes values, $\overline{\mathbf{M}}$, such that for each element $\mathbf{m}$ of $\overline{\mathbf{M}}$, $\left\|\mathbf{m} - \mathbf{d}\right\|_{\rho} = 0$;

if $\overline{\mathbf{M}}$ is empty, then replace each missing entry in $\mathbf{d}$ with corresponding entries in $\mathbf{n}$; and

if $\overline{\mathbf{M}}$ is not empty, then replace each missing entry in $\mathbf{d}$ with the average value of the corresponding position of all the elements in $\overline{\mathbf{M}}$.

21. A method according to any one of the preceding claims, wherein the data representation includes a knowledge filter.

22. A method of computer data analysis using neural networks, the method including:

generating a data set $\overline{\mathbf{D}}$, the data set including a plurality of attributes and a plurality of data set nodes;

initializing the data set, initializing the data set including:

calculating an autocorrelation matrix, $\aleph$ over the input data set $\overline{\mathbf{D}}$,

where $\aleph = \frac{1}{card(\overline{\mathbf{D}})} \sum_{\forall \mathbf{d} \in \overline{\mathbf{D}}} \mathbf{d} \cdot \mathbf{d}^T$;

finding two longest eigenvectors of $\aleph$, e_1 and e_2, where $|e_1| > |e_2|$; and

initializing vector values of each element of a data representation F by spanning it with element values of the eigenvectors;

generating a data representation using a training algorithm, wherein the training algorithm includes growing the data set, growing the data set including:

finding K_q for each of the data set nodes, where K_q is the node with the highest average quantization error, $\arg\max_q \{\overline{q}(t)_{K_q}\}$ for each of the data set nodes, where $\overline{q}(t)_{K_q} = \frac{1}{t-1}\sum_{t=1}^{t=t-1} q(t)_{K_q}$ is the average quantization error for node q, where:

$$K_x = \arg\max_x \{\|K_q - K_{<r(q)-1,c(q)>}\|, \|K_q - K_{<r(q)+1,c(q)>}\|\}$$

$$K_y = \arg\max_y \{\|K_q - K_{<r(q),c(q)-1>}\|, \|K_q - K_{<r(q),c(q)+1>}\|\}$$

if $\|K_y - K_c\| < \|K_x - K_c\|$ then

$n_r = r(y)$ if $r(y) < r(c)$, else $n_r = r(c)$; and

$n_c = c(y)$;

else $n_r = r(y)$; $n_c = c(x)$ if $c(x) < c(c)$, else $n_c = c(c)$;

inserting a new row and column after row n_r and column n_c;

interpolate new attribute values for the newly inserted node vectors using: $K_{<r,n_c>} = (K_{<r,n_c-1>} + K_{<r,c_n+1>})\frac{\alpha}{2}$ and

$$K_{<n_r,c>} = (K_{<n_r-1,c>} + K_{<n_r+1,c>})\frac{\alpha}{2}, \text{ where } \alpha \in U(0,1);$$

performing convergence testing, wherein convergence testing checks for convergence of the training algorithm, and wherein the training algorithm is repeated until convergence of the training algorithm occurs; and

displaying one or more subsets of the data set using the data representation.

23. A method according to claim 22, wherein initializing the vector values further includes:

$$F_{<F_R,0>} := 0$$

$$F_{<F_R,F_C>} := \mathbf{e}_1$$

$$F_{<1,F_C>} := \mathbf{e}_1 + \mathbf{e}_2$$

$$F_{<F_R,F_C>} := \mathbf{e}_2$$

$$\forall c \in [2, F_C - 1],\ F_{<1,c>} := \frac{F_C}{F_C - c} F_{<1,F_C>} + \frac{F_C - c}{F_C} F_{<1,1>}$$

$$\forall c \in [2, F_C - 1],\ F_{<R_C,c>} := \frac{c}{F_C} F_{<F_R,F_C>} + \frac{F_C - c}{F_C} F_{<F_R,1>}$$

$$\forall r \in [2, F_R - 1],\ F_{<r,1>} := \frac{c}{F_R} F_{<F_R,1>} + \frac{F_R - r}{F_R} F_{<1,1>}$$

$$\forall r \in [2, F_R - 1],\ F_{<r,F_C>} := \frac{r}{F_R} F_{<F_R,F_C>} + \frac{F_R - r}{F_R} F_{<r,F_R>}$$

$$\forall r \in [2, F_R - 1],\ \forall c \in [2, F_C - 1],\ F_{<r,c>} := \frac{c}{F_C} F_{<r,F_C>} + \frac{F_C - c}{F_C} F_{<r,1>}.$$

24. A method according to claim 22 or 23, wherein the training algorithm further includes:

$t = t + 1;$

$\forall \mathbf{d} \in \overline{\mathbf{D}}$

if ($t < 50$ or *afterGrow*)

$$\wp_{\mathbf{d}} = \underset{<r,c>,\forall r \in [1,F_R],\forall c \in [1,F_C]}{\arg\min} \left\| \mathbf{d} - F_{<r,c>} \right\|_{\rho}$$

$afterGrow = false$

else

$\wp_{\mathbf{d}} = FindSCWS(\mathbf{d})$

call function: $FindNeighborhoodPatterns(\overline{\wp})$

call function: $BatchUpdateMatchVectors$

$$q(t) = \frac{1}{card(\overline{\mathbf{D}})} \sum_{t} \left(\left\| \mathbf{d} - F_{\wp \mathbf{d}} \right\| \rho \right).$$

if ($MayGrow(t)$ and $t < t_{\max}$), call function: $GrowKF$.

25. A method according to claim 22, 23, or 24, wherein performing convergence testing includes testing condition $q(t) < Q_e$.

26. A method according to any one of claims 22 to 25, wherein displaying one or more subsets of the data set includes using a composite view

to view multiple attributes, wherein an additional attribute image is created, the additional attribute image displaying a union of a selected set of attributes.

27. A method according to claim 26, wherein using a composite view further includes:

constructing an attribute matrix; and

selecting a highest value for each attribute value from the selected set of attributes.

28. A method according to any one of claims 22 to 27, wherein displaying one or more subsets of the data set includes uses a range filter to select regions on the data representation and filter out nodes based on defined value ranges .

29. A method according to any one of claims 22 to 28, wherein displaying one or more subsets of the data set includes using a zooming function, wherein the zooming function includes:

making a selection of nodes to form a base reference of interest;

defining a set of data records from a second data set;

matching the second data set to the data representation;

flagging all records that are linked to the matched region; and

generating a second data representation using the flagged records.

30. A method according to any one of claims 22 to 29, wherein displaying one or more subsets of the data set includes using visual scaling, changing the minimum and maximum values used to calculate a colour progression used to visualize at least one of the plurality of attributes, and re-interpolating the active colour ranges over the new valid range of attribute values.

31. A method according to any one of claims 22 to 30, wherein displaying one or more subsets of the data set includes using a labeling engine to:

linking attribute columns in an input file to attributes in the data representation;

selecting attributes from the input file to be used for labelling;

determining with which row and column each row in the input file is associated; and

placing labels on the data representation.

32. A method according to any one of claims 22 to 31, wherein displaying one or more subsets of the data set includes using an advanced search function to:

read a set of data records from a data source;

match attribute columns from the set of data records to attributes in the data representation; and

display a list of all records that are associated with nodes that are part of the active selection on the data representation.

33. A method according to any one of claims 22 to 32, wherein displaying one or more subsets of the data set includes using equal distance averaging (EDA), wherein equal distance averaging includes:

finding the node vector, $\mathbf{n}$, in the data representation that most closely represents the input data vector, $\mathbf{d}$:

$\mathbf{n} = \arg\min_{K_{<r,c>}} \left\{ \left\| K_{<r,c>} - \mathbf{d} \right\|_{\rho} \right\},\ \forall r \in [1, K_R], \forall c \in [1, K_C]$; and

replacing missing entries in $\mathbf{d}$ with the corresponding entries from $\mathbf{n}$.

34. A method according to claim 33, wherein the equal distance averaging further includes:

building a list of the data representation nodes values, $\overline{\mathbf{M}}$, such that for each element $\mathbf{m}$ of $\overline{\mathbf{M}}$, $\left\| \mathbf{m} - \mathbf{d} \right\|_{\rho} = 0$;

if $\overline{\mathbf{M}}$ is empty, then replace each missing entry in $\mathbf{d}$ with corresponding entries in $\mathbf{n}$; and

if $\overline{\mathbf{M}}$ is not empty, then replace each missing entry in $\mathbf{d}$ with the average value of the corresponding position of all the elements in $\overline{\mathbf{M}}$.

35. A method according to any one of claims 22 to 34, wherein the data representation is a knowledge filter.

36. A method according to any one of the preceding claims wherein the data representation includes a latent model of the data set.

37. A system for performing data analysis using neural networks, the system including:

one or more processors;

one or more memories coupled to the one or more processors; and

program instructions stored in the one or more memories, the one or more processors being operable to execute the program instructions, the program instructions including:

generating a data representation using a data set, the data set including a plurality of attributes, wherein generating the data representation includes:

modifying the data set using a training algorithm, wherein the training algorithm includes growing the data set; and

performing convergence testing, wherein convergence testing checks for convergence of the training algorithm, and wherein the modifying of the data set is repeated until convergence of the training algorithm occurs; and

displaying one or more subsets of the data set using the data representation.

38. A system according to claim 37, wherein performing convergence testing includes testing condition $q(t) < Q_e$.

39. A system according to claim 37 or 38, wherein the data set includes a plurality of data set nodes, and wherein growing the data set includes:

finding K_q for each of the data set nodes, where K_q is the node with the highest average quantization error, $\arg\max_q \{\overline{q}(t)_{K_q}\}$ for each of the data set nodes, where $\overline{q}(t)_{K_q} = \frac{1}{t-1}\sum_{t=1}^{t=t-1} q(t)_{K_q}$ is the average quantization error for node q, where:

$$K_x = \arg\max_x \{\|K_q - K_{<r(q)-1,c(q)>}\|, \|K_q - K_{<r(q)+1,c(q)>}\|\}$$

$$K_y = \arg\max_y \{\|K_q - K_{<r(q),c(q)-1>}\|, \|K_q - K_{<r(q),c(q)+1>}\|\}$$

if $\|K_y - K_c\| < \|K_x - K_c\|$ then

$n_r = r(y)$ if $r(y) < r(c)$, else $n_r = r(c)$; and

$n_c = c(y)$;

else $n_r = r(y)$; $n_c = c(x)$ if $c(x) < c(c)$, else $n_c = c(c)$;

inserting a new row and column after row n_r and column n_c; and

interpolating new attribute values for the newly inserted node vectors using: $K_{<r,n_c>} = \left(K_{<r,n_c-1>} + K_{<r,c_n+1>}\right)\frac{\alpha}{2}$ and

$K_{<n_r,c>} = \left(K_{<n_r-1,c>} + K_{<n_r+1,c>}\right)\frac{\alpha}{2}$, where $\alpha \in U(0,1)$.

40. A system according to claim 37, 38, or 39, wherein the training algorithm further includes:

$t = t + 1$;

$\forall \mathbf{d} \in \overline{\mathbf{D}}$;

if ($t < 50$ or *afterGrow*)

$$\wp_{\mathbf{d}} = \underset{<r,c>, \forall r \in [1,F_R], \forall c \in [1,F_C]}{\arg\min} \left\| \mathbf{d} - F_{<r,c>} \right\|_{\rho}$$

$afterGrow = false$

else

$\wp_{\mathbf{d}} = FindSCWS(\mathbf{d})$

call function: $FindNeighborhoodPatterns(\overline{\wp})$

call function: $BatchUpdateMatchVectors$

$$q(t) = \frac{1}{card(\overline{\mathbf{D}})} \sum_{t} \left(\left\| \mathbf{d} - F_{\wp \mathbf{d}} \right\| \rho \right);$$ and

if ($MayGrow(t)$ and $t < t_{\max}$), call function: $GrowKF$.

41. A system according to any one of claims 37 to 40, wherein the program instructions further include: displaying one or more subsets of the data set includes using a composite view to view multiple attributes, wherein an additional attribute image is created, the additional attribute image displaying a union of a selected set of attributes.

42. A system according to any one of claims 37 to 41, wherein the program instructions further include:

constructing an attribute matrix; and

selecting a highest value for each attribute value from the selected set of attributes.

43. A system according to any one of claims 37 to 42, wherein displaying one or more subsets of the data set includes uses a range filter to select regions on the data representation and filter out nodes based on defined value ranges .

44. A system according to any one of claims 37 to 43, wherein displaying one or more subsets of the data set includes using a zooming function, wherein the wherein the program instructions further include:

making a selection of nodes to form a base reference of interest;

defining a set of data records from a second data set;

matching the second data set to the data representation;

flagging all records that are linked to the matched region; and

generating a second data representation using the flagged records.

45. A system according to any one of claims 37 to 44, wherein displaying one or more subsets of the data set includes using visual scaling, wherein the wherein the program instructions further include:

changing the minimum and maximum values used to calculate a colour progression used to visualize at least one of the plurality of attributes;

and re-interpolating the active colour ranges over the new valid range of attribute values.

46. A system according to any one of claims 37 to 45, wherein displaying one or more subsets of the data set includes using a labeling engine, wherein the program instructions further include:

linking attribute columns in an input file to attributes in the data representation;

selecting attributes from the input file to be used for labelling;

determining with which row and column each row in the input file is associated; and

placing labels on the data representation.

47. A system according to any one of claims 37 to 46, wherein displaying one or more subsets of the data set includes using an advanced search engine, wherein the program instructions further include:

displaying one or more subsets of the data set includes using an advanced search function to:

read a set of data records from a data source;

match attribute columns from the set of data records to attributes in the data representation; and

display a list of all records that are associated with nodes that are part of the active selection on the data representation.

48. A system according to any one of claims 37 to 47, wherein displaying one or more subsets of the data set includes using equal distance averaging (EDA), wherein the program instructions further include:

finding the node vector, $\mathbf{n}$, in the data representation that most closely represents the input data vector, $\mathbf{d}$:

$\mathbf{n} = \arg\min_{K_{<r,c>}} \left\{ \left\| K_{<r,c>} - \mathbf{d} \right\|_p \right\}, \ \forall r \in [1, K_R], \forall c \in [1, K_C]$; and

replacing missing entries in $\mathbf{d}$ with the corresponding entries from $\mathbf{n}$.

49. A system according to claim 48, wherein the program instructions further include:

building a list of the data representation nodes values, $\overline{\mathbf{M}}$, such that for each element $\mathbf{m}$ of $\overline{\mathbf{M}}$, $\left\| \mathbf{m} - \mathbf{d} \right\|_p = 0$;

if $\overline{\mathbf{M}}$ is empty, then replace each missing entry in $\mathbf{d}$ with corresponding entries in $\mathbf{n}$; and

if $\overline{\mathbf{M}}$ is not empty, then replace each missing entry in $\mathbf{d}$ with the average value of the corresponding position of all the elements in $\overline{\mathbf{M}}$.

50. A system according to any one of claims 37 to 49, wherein the data representation includes a knowledge filter.

51. A system according to any one of claims 37 to 50, wherein the data representation includes a latent model of the data set.

52. A computer program product for computer data analysis using neural networks, the computer program product including:

computer-readable program code for generating a data representation using a data set, the data set including a plurality of attributes, wherein generating the data representation includes:

modifying the data set using a training algorithm, wherein the training algorithm includes growing the data set; and

performing convergence testing, wherein convergence testing checks for convergence of the training algorithm, and wherein the modifying of the data set is repeated until convergence of the training algorithm occurs; and

computer-readable program code for displaying one or more subsets of the data set using the data representation.

53. A computer program product according to claim 52, wherein the data set includes a plurality of data set nodes, and the computer program product further including computer-readable program code for growing the data set including:

finding K_q for each of the data set nodes, where K_q is the node with the highest average quantization error, $\arg\max_q \{\overline{q}(t)_{K_q}\}$ for each of the data set nodes, where $\overline{q}(t)_{K_q} = \frac{1}{t-1}\sum_{t=1}^{t=t-1} q(t)_{K_q}$ is the average quantization error for node q, where:

$$K_x = \arg\max_x \{\|K_q - K_{<r(q)-1,c(q)>}\|, \|K_q - K_{<r(q)+1,c(q)>}\|\}$$

$$K_y = \arg\max_y \{\|K_q - K_{<r(q),c(q)-1>}\|, \|K_q - K_{<r(q),c(q)+1>}\|\}$$

if $\|K_y - K_c\| < \|K_x - K_c\|$ then

$n_r = r(y)$ if $r(y) < r(c)$, else $n_r = r(c)$; and

$n_c = c(y)$;

else $n_r = r(y)$; $n_c = c(x)$ if $c(x) < c(c)$, else $n_c = c(c)$;

inserting a new row and column after row n_r and column n_c; and

interpolating new attribute values for the newly inserted node vectors using: $K_{<r,n_c>} = \left(K_{<r,n_c-1>} + K_{<r,c_n+1>}\right)\frac{\alpha}{2}$ and

$K_{<n_r,c>} = \left(K_{<n_r-1,c>} + K_{<n_r+1,c>}\right)\frac{\alpha}{2}$, where $\alpha \in U(0,1)$.

54. A computer program product according to claim 52 or 53, wherein the data representation includes a latent model of the data set.

55. An apparatus for performing data analysis using neural networks, the apparatus including:

means for representing a data set, the data set including a plurality of attributes;

means for generating the representation means using the data set, wherein generating the representation means includes:

modifying the data set using a training algorithm, wherein the training algorithm includes growing the data set; and

performing convergence testing, wherein convergence testing checks for convergence of the training algorithm, and wherein the modifying of the data set is repeated until convergence of the training algorithm occurs; and

means for displaying one or more subsets of the data set using the modified data representation.

56. An apparatus according to claim 55, wherein the representation means includes a latent model of the data set.

57. A method of computer data analysis using neural networks substantially as herein described with reference to the accompanying drawings.

58. A system for performing data analysis using neural networks substantially as herein described with reference to the accompanying drawings.

59. A computer program product for computer data analysis using neural networks substantially as herein described with reference to the accompanying drawings.

60. An apparatus for performing data analysis using neural networks substantially as herein described with reference to the accompanying drawings.